

# Gradient-Angular-Features for Word-Wise Video Script Identification

<sup>a</sup>Palaiahnakote Shivakumara, <sup>b</sup>Nabin Sharma, <sup>c</sup>Umapada Pal, <sup>b</sup>Michael Blumenstein and <sup>d</sup>Chew Lim Tan

<sup>a</sup>Faculty of Computer Science, University of Malaya, Kuala Lumpur, Malaysia

<sup>b</sup>Griffith University, Gold Coast, Queensland, Australia

<sup>c</sup>Computer Vision and Pattern Recognition Unit, Indian Statistical Institute, India

<sup>d</sup>School of Computing, National University of Singapore, Singapore

**Abstract**—Script identification at the word level is challenging because of complex backgrounds and low resolution of video. The presence of graphics and scene text in video makes the problem more challenging. In this paper, we employ gradient angle segmentation on words from video text lines. This paper presents new Gradient-Angular-Features (GAF) for video script identification, namely, Arabic, Chinese, English, Japanese, Korean and Tamil. This work enables us to select an appropriate OCR when the frame has words of multi-scripts. We employ gradient directional features for segmenting words from video text lines. For each segmented word, we study the gradient information in effective ways to identify text candidates. The skeleton of the text candidates is analyzed to identify Potential Text Candidates (PTC) by filtering out unwanted text candidates. We propose novel GAF for the PTC to study the structure of the components in the form of cursiveness and softness. The histogram operation on the GAF is performed in different ways to obtain discriminative features. The method is evaluated on 760 words of six scripts having low contrast, complex background, different font sizes, etc. in terms of the classification rate and is compared with an existing method to show the effectiveness of the method. We achieve 88.2% average classification rate.

**Keywords**—Video words, Gradient words, Text candidates, Potential text candidates, Gradient-Angular-Features, Video script identification

## I. INTRODUCTION

In countries like Singapore, India and Malaysia, where multi-languages are used as official languages due to diversity in culture and religion, we may see video frames containing multi-scripts. As a result, it is a big challenge for researchers in the field of image processing and pattern recognition to understand multi-script video frames because the current OCR engines are designed for recognizing frames having a single script rather than multiple scripts. In addition, there is no universal OCR that recognizes more than one script so far as per our knowledge. Therefore, it is necessary to identify the scripts in order to choose an appropriate OCR for a frame containing different scripts. However, low resolution, complex background, arbitrary orientation, font variation, font size variation etc of video text make this problem more complex and challenging [1-2].

There are methods for text detection, region detection and block detection in video [3-5], which can be classified as connected component-based, texture-based, and gradient and edge-based methods. Although these methods achieve good accuracy for text detection irrespective of fonts, font sizes, types

of text and orientation, they do not have the ability to differentiate multiple scripts in a frame because the goal of these methods is to detect text lines regardless of scripts in the video [3-5]. Hence, the text detection methods cannot be used for video script identification.

To the best of our knowledge, the work on video script identification is not much reported in the literature. For instance, the method in [1] which takes text lines as input and uses statistical and texture features with k-nearest neighbor classifier to identify Latin and Ideographic text in images and videos [1]. This method works well for high contrast English and Chinese text line images but not for other scripts at the word level because the texture features are extracted based on zonalization of text line images and the performance of the method is dependent on number of nearest neighbors used for classification. These features may not be good enough to identify video scripts at the word level. New features, namely, smoothness and cursiveness based on text lines without a classifier are recently proposed in [6] for video script identification. The scope of the work is limited to only two scripts, English and Chinese or English and Tamil. To overcome this limitation, Danni et al. [7] have proposed a method based on spatial-gradient-features to identify the script in a frame at block level. This method works well for multi-script frames but not a single frame containing multi-scripts because it expects any one of the blocks out of 16 to satisfy the condition that classifies the particular script. This is the major concern of this method. In addition, fixed block size cause loss of information. These factors motivated us to propose a method for video script identification at the word level, which works for the frame containing multiple scripts of words.

Identification of different scripts at the text line level, word level and block level is not new for the document analysis field [8-11] where we can see plenty of methods for multiple script identification. For instance, an overview of script identification methodologies based on structure and visual appearance can be found in [8] where the proposed methods work well for plain and high resolution camera-based images but not for video frames which usually have contrast, background, different font shapes, different font sizes, orientations, illumination variations and distortion. For the word level, there are methods [12-16] where the methods require good segmentation of words from the text lines. Otherwise, the segmentation error affects the script identification step. Script identification by studying the character shape can also be found in [17, 18]. Again the performance of the method depends on how good the segmentation method is. The method for script identification and orientation detection for



Indian text images has been proposed by Ghosh and Chaudhuri [19]. Since the features are derived from the connected component analysis of the characters, the performance of the method is good if the shape of the character is preserved. Recently, Sharma et al. [20] proposed word-wise script identification in video at word level. This method proposes Zernike moments, Gabor features and gradient features with SVM classifier. The method considers only Indian scripts and English but not like scripts Chinese, Japanese, Korean etc. where powerful method is required.

From the above discussion on both camera and video documents, it is observed that the main objective of the methods is identification of scripts in high contrast and plain background images. Therefore, the methods in document analysis may not work for the video script identification because video scripts suffer from low contrast and complex background which cause disconnections, loss of shapes and information. Hence, we can infer that the problem of script identification in video, in particular the identification of Arabic, Chinese, English, Japanese, Korean and Tamil remains unsolved problem and is challenging.

Thus, in this paper, we present a novel method for video script identification at the word level by considering six well known scripts in the world. Inspired by the features presented in [7] for extracting text components from video blocks by performing histogram operation on gradient blocks, we explore the gradient angular features at the word level rather than block level in this work to obtain text candidates. We are motivated to propose angular features in a gradient domain because it is shown that the angular features are useful for detecting seals in complex backgrounds in [21]. However, the method uses angle information determined by three consecutive pixels of the contour of external and internal region of characters. The success of the method depends on the connectivity of the contour and the shape of the character. Therefore, these features cannot be used directly to tackle the video script identification problem. Hence, we extract angular features in a novel way to solve the video script identification problem in this work.

## II. PROPOSED METHODOLOGY

We have developed several methods for segmenting words in text lines in video frames. We use one of our developed methods [22] for segmenting words, which detects text lines of arbitrary orientation of text from video first then it segments the words from the text line. The word segmentation method works well for different scripts except Chinese, Japanese and Korean because it is hard to define space between words as we define for English and other scripts. In addition, it works for arbitrary orientation of texts. The method in [22] uses four gradient directional features to produce the respective pixel candidate images. The text candidate image is then computed by performing a union operation on the pixel candidate images. A nearest neighbor criterion is used to find the text component sequence in a text line. The method then uses a two-stage approach for word segmentation. In the first stage, the distances between the components are clustered using K-means clustering with  $k=2$ , to determine the probable word and non-word spacing. The first stage does the initial word segmentation allowing under-segmentation and avoiding over-segmentation. Whereas, in the second stage, the word segmentation results obtained from the first stage are verified whether they are under-segmented and are segmented further if required. Figure 1(a)-(c) shows the method segmenting the words successfully for the different Arabic, English and Tamil script text lines. However, for

Chinese, Japanese and Korean, the method does not segment the words from the lines due to uncertainty in defining the space between the words. Therefore, in this work, we segment words for Arabic, English and Tamil automatically and we segment the words for Chinese, Japanese and Korean manually (refer Figure 1(d)-(f)).

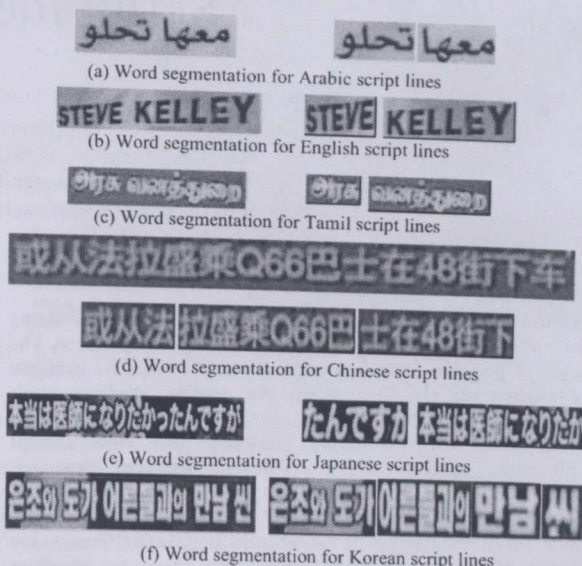


Fig. 1. Illustrating word segmentation by the method [22] for Arabic, English, Tamil, Chinese, Japanese and Korean script lines.

The algorithmic steps for video script identification for the segmented words are described in subsequent sections.

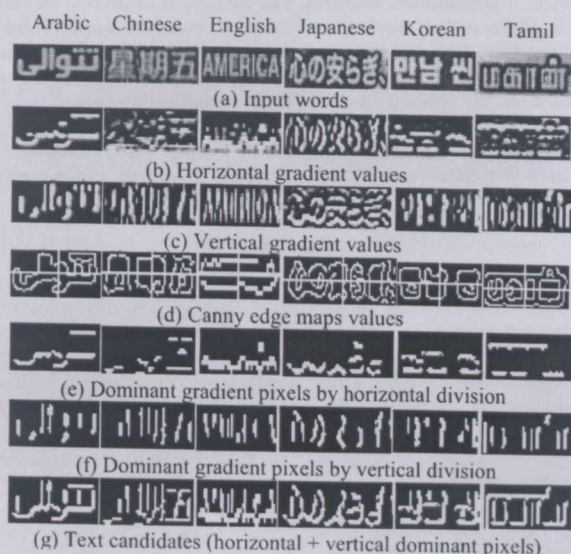


Fig. 2. Intermediate results for text components

### A. Gradient Histogram for Text Candidates

For each pixel in the word as shown in Figure 2(a), the method obtains gradient values by convolving the Sobel horizontal and



vertical masks over the word image to increase the contrast as shown in Figure 2(b) and (c), respectively. It is noted from the literature that the gradient information is useful for video text pixel classification as it gives high values for high contrast pixels (text) and low values for low contrast pixels (non-text) [4]. In order to select dominant text pixels (edge pixels) which usually have high gradient values, the method divides the horizontal gradient word at the centroid horizontally, which results in two equal parts namely, the upper part and the lower part. The centroid is computed based on edge pixels in Canny edge map of the input block as shown in Figure 2(d). For the upper and the lower parts of gradient blocks, we plot a histogram to find gradient values which give the highest peak bin in the histogram and these values are considered as dominant values of text pixels. The dominant text pixels are represented as white pixels and other gradient values are represented as black pixels as shown in Figure 2(e).

In the same way, the method divides the vertical gradient word image vertically at the centroid to obtain the left part and the right part of the gradient word. The same histogram criterion is used to select dominant pixels from both the right and the left parts of the gradient word as shown in Figure 2(f). Then we combine the horizontal and vertical dominant pixels to restore the text information which we call text candidates as shown in Figure 2(g) where one can notice almost all pixels of text are restored and structure of the character can be seen. The way the method divides a gradient word image to identify dominant text pixels ensures that most text pixels do not escape detection. Therefore, we can conclude that this step works well for low contrast images like video images. Figure 3 shows how horizontal and vertical division are performed for the gradient word image, histogram operation on each division to get dominant pixel with the help of highest peak and final text candidates by combining dominant pixel of horizontal and vertical divisions.

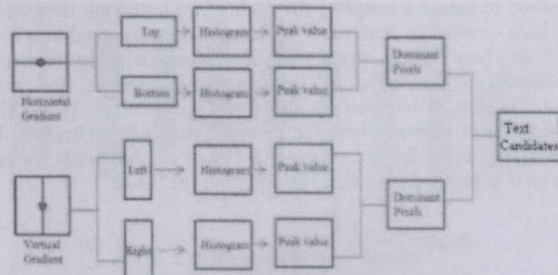


Fig. 3. Flow diagram for text component selection using gradient histogram.

### B. Potential Text Candidate Selection

To study the structure of text candidates, we obtain skeletons using a skeletonization function, which results in single pixel width text candidates as shown in Figure 4(a). In addition, it also helps in extracting angular features accurately to study the structure of the text candidates. The method eliminates unwanted text candidates that have small area (i.e. the number of pixels in the text candidates). This results in Potential Text Candidate (PTC) as shown in Figure 4(b). For these PTC, we extract Gradient-Angular-Features (GAF) which will be discussed in the next subsection.

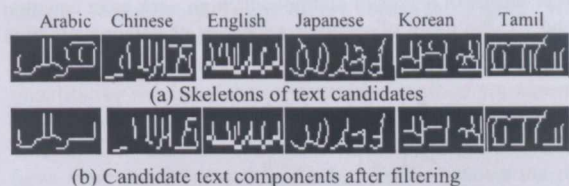


Fig. 4. Candidate text components

### C. Gradient-Angular-Features for Words

It can be seen from Figure 4(b) that the PTC in Arabic, English scripts do not show much cursiveness compared to the PTC in Chinese, Japanese, Korean and Tamil. This is valid because Arabic and English candidates involve much horizontal and vertical stroke information while other little horizontal and vertical stroke information and more cursive stroke information. This gives clue that there is more softness (less cursiveness including straightness) in case of Arabic and English PTC while there is more cursiveness in case of other script PTC. In addition, we can also notice from Figure 4(b) that the PTC are broken into several small segments due to low resolution and complex background. Thus in this work, we propose new Gradient-Angular-Features (GAF) that are invariant to disconnections, broken segments, font size variation, font variation since the angle information remains unaffected by the broken segments and other problems to study the softness and cursiveness of the components in PTC. In addition, the way the method normalizes both the extracted global and local features makes these features invariant to font variation. Nevertheless, the proposed features are sensitive to orientation. In this way, the proposed features are effective and are different from the literature to identify the word-wise video scripts. The detailed steps for the feature extraction are as follows. The method moves a span of five pixels over the contour pixel by pixel along the contour of a component in PTC in the word as illustrated in Figure 5 where (a) is the input Arabic PTC, (b)-(e) are the zoomed version of the first component in (a) which show how the sliding window covering five pixels move one pixel at a time (red color) along the contour. Let the first pixel be P1 and fifth pixel be P5, angle of the short contour in degrees is calculated as given in equation (1). The method extracts gradient-angular-features for the components in the PTC of the word image in this way. For the GAF, we perform the histogram operation to find values in the high, the low peak and all the peak bins at both component and word level. In order to make features invariant to the number of components, the mean and the median of those bins are computed in different way to study the softness and cursiveness of the components in the scripts. In total, we extract eight features (F1-F8) to identify the six scripts as in the following.

$$Angle = 180^\circ + \tan^{-1} \frac{P1_x - P5_x}{P1_y - P5_y} \quad (1)$$

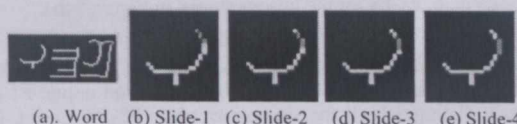


Fig. 5. Moving sliding window along contour

For a word with M components, let *CompDP* be the set of all angles in the high peak are computed as in equation (1). For a sorted (by median) set of N angles, we can calculate F1 as the



mean of means of angles of components high peak as in equation (2) and F2, the mean of medians of angles of components high peak as in equation (3).

$$F1 = \frac{1}{M} \sum_{i=1}^M \left[ \frac{1}{N} \sum_{j=1}^N CompDP_j \right]_i \quad (2)$$

$$F2 = \frac{1}{M} \sum_{i=1}^M [CompDP_{(N+1)/2}]_i \quad (3)$$

F3, the mean of means of angles of component low peak and F4, the mean of medians of angles of component low peak are calculated as above except that *CompDP* is replaced by *CompLP* which is the set of all angles in the low peak.

Examining the histogram at the word level, there are many bins that satisfy the low peak criteria, F5 calculates the median of *WordLP* for a sorted (by median) set of N angles in low peak bins at the word level as in equation (4).

$$F5 = WordLP_{(N+1)/2} \quad (4)$$

F6 finds the mean of means of *CompAngle*, the set of all angles in the component as in equation (5).

$$F6 = \frac{1}{M} \sum_{i=1}^M \left[ \frac{1}{N} \sum_{j=1}^N CompAngle_j \right]_i \quad (5)$$

Finally for the set of all angles at the word level, *WordAngle*, we find the mean and median of *WordAngle* as our F7 and F8, respectively for a sorted (by median) set of N angles as defined in equation (6) and (7).

$$F7 = \frac{1}{N} \sum_{i=1}^N WordAngle_i \quad (6), \quad F8 = WordAngle_{(N+1)/2} \quad (7)$$

It is noted that the features F1-F6 extract local information of the components and the Features F7 and F8 extract global information of the components. The combination of local and global angle information of word image help us to obtain the distinct features for identifying the scripts accurately.

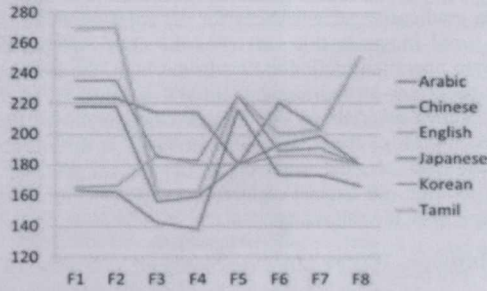


Fig. 6. F1-F8 for the images shown in Figure 3 (b).

The usefulness of the features F1-F8 for the image shown in Figure 4(b) is illustrated in Figure 6 where we can notice F1 and F2 clearly distinguish all the six scripts while F3-F8 do not. It is also observed from Figure 6 that the line graphs of Arabic and English appear as soft compared to the line graphs of the other scripts. Therefore, it is inferred that the Arabic and English do not

have much cursiveness compared to the other scripts due to the nature of the scripts.

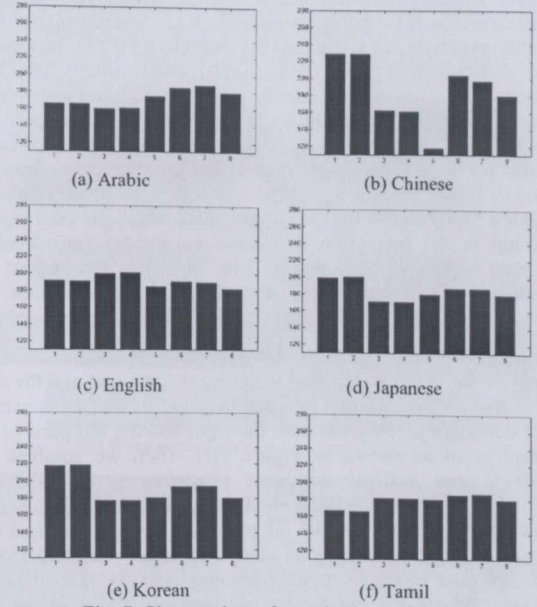


Fig. 7. Six templates for script identification

#### D. Template Creation

It is found that the extracted features in the previous subsection are robust and effective for obtaining distinct features for the scripts. Therefore, in this subsection, we propose a simple method to create a template for each script by taking the average of each respective feature of 50 samples chosen randomly from our database rather than proposing classifiers, to find high probability. The idea of constructing templates for script identification is inspired by the work presented in [12, 13] in which script identification for camera images is addressed. The average of eight feature vector (F1-F8) of the 50 words for each script is computed as defined in equation (8) below.

$$Avg(FV) = \frac{1}{50} \sum_{i=1}^{50} FV_i \quad (8)$$

Here, FV is the vector consisting of F1-F8 features. This gives six templates (vectors) for the six scripts containing eight average features in each template. For a given word, the method extracts FV and compares them with the six templates to find the minimum Euclidean distance to classify the word into a particular class. This procedure gives a confusion matrix for the six scripts. The sample templates for the scripts Arabic, Chinese, English, Japanese, Korean and Tamil can be seen respectively in Figure 7(a)-(f) where all the six templates have distinct features. This shows that constructed templates are good enough to classify the six scripts with good classification rate. The Euclidean distance for classification is defined as follows. For a word with extracted feature vector FV, the set of Euclidean distance  $ED = \{ED_1, ED_2, \dots, ED_6\}$  between FV and the set of template vector  $T = \{T_1, T_2, \dots, T_6\}$  is given by,

$$ED_i = \sqrt{\sum_{r=1}^8 (FV_r - T_{ir})^2} \quad (9)$$



Where  $d$  is the dimension, in this case,  $d = 8$  since there are 8 features. The classified script is given by  $S \in \{1, 2 \dots 6\}$  where

$$ED_S = \min(ED)$$

### III. EXPERIMENTAL RESULTS

Since there is no standard dataset or benchmark dataset available publicly for word-wise video script identification, we created our own dataset from different sources such as sports news, weather news, entertainment video etc to show that the proposed method works well for different varieties of video words. Our dataset include 170 Arabic, 120 Chinese, 110 English, 100 Japanese, 160 Korean and 100 Tamil words. In total, 760 words are used for the purpose of experimentation. To evaluate the performance of the method, we consider classification rate as a measure and we present a confusion matrix containing classification rate (diagonal elements)/misclassification rate (in column) for each script.

#### A. Individual Feature Analysis

The aim of this experiment is to find the contribution of each feature in terms of average classification rate (average of diagonal elements of confusion matrices of each feature). It is observed from Figure 8 that each feature contributes to the classification of scripts. Especially, feature-3 and feature-4 perform better compared to other features as they achieve a higher average classification rate. We combine these eight features for classification of the six scripts. This analysis helps us to revise the features extraction method to improve the accuracy and efficiency.

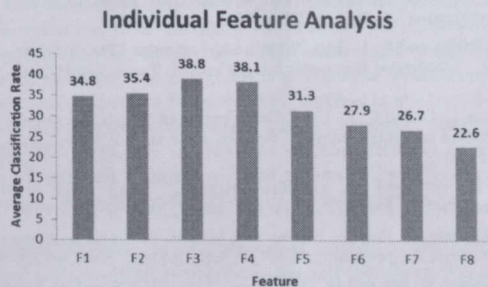


Fig. 8. Contribution of each feature

#### B. Experimental Results on Word Segmentation

We also evaluate the segmentation method [22] for segmenting words from different script lines in terms of recall, precision and F-measure. It is noted that the segmentation method works well for Arabic, English and Tamil but not Chinese, Japanese and Korean because of unambiguity in defining exact space between the words. In case of Arabic, English and Tamil, we can define space as the spacing between the words is greater than characters and we can see regular spacing between the words. This is not true for Chinese, Japanese and Korean script lines. Therefore, we calculate recall, precision and F-measure only for Arabic, English and Tamil scripts and for Chinese, Japanese and Korean script lines, we segment manually. Sample results of the segmentation method for English, Tamil and Arabic are

shown respectively from Figure 9(a)-(c). Figure 9(a)-(c) shows that the segmentation method segments the words of different orientation, contrast and fonts successfully. The quantitative results of the segmentation method are reported in Table I where we the method gives better results for Arabic compared to English and Tamil because English and Tamil have more cursiveness than Arabic. Table I shows the results are encouraging and promising as all scripts get more than 75% F-measure.

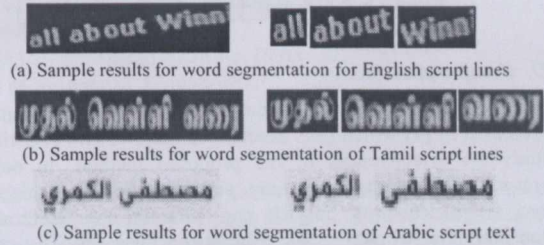


Fig. 9. Sample results of word segmentation by the method [22]

TABLE I. WORD SEGMENTATION RESULTS

Dataset	Recall	Precision	f-measure
English	0.77	0.76	0.77
Tamil	0.76	0.75	0.76
Arabic	0.79	0.78	0.79

#### C. Experimental Results for Script Identification

Sample words of the six scripts are shown in Figure 8 where Figure 10(a)-(f) show Arabic, Chinese, English, Japanese, Korea and Tamil script, respectively. It is noticed from Figure 10 that we consider varieties of words including complex background, low resolution text, different fonts, fonts sizes etc to show that the proposed method work well for different situations. The result of GAF for classification of the six scripts is reported in Table II where classification rate for all the six scripts is promising. However, for Tamil, the method gives low classification rate compared to other scripts as the features may overlap with the features of Chinese and other scripts. Despite low accuracy for Tamil, we achieve an 88.2% average classification rate (average of diagonal elements in Table II).

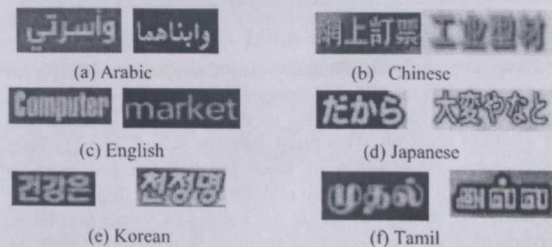


Fig. 10. Sample blocks for six scripts from our database



TABLE II. CONFUSION MATRIX OF THE PROPOSED METHOD (IN %)  
(A: Arabic, C: Chinese, E: English, J: Japanese, K: Korean, T: Tamil)

Scripts	A	C	E	J	K	T
Arabic	<b>89.4</b>	3.3	5.5	3	1.9	2
Chinese	4.1	<b>90.8</b>	2.7	2	2.5	6
English	2.4	2.5	<b>89.1</b>	2	0.6	3
Japanese	0.6	0	0	<b>89</b>	2.5	4
Korean	1.2	2.5	0	3	<b>89.4</b>	3
Tamil	2.4	0.8	2.7	1	3.1	<b>82</b>

#### D. Comparative Study

The proposed method is compared with the existing method presented in [1] which uses descriptive features and classifiers to study the effectiveness of the proposed method. In order to present a fair comparative study, since this work considers only two scripts such as English and Chinese, our method also considers the two scripts for comparison purposes. The existing method considers text lines as input for script identification while the proposed method considers words as input, therefore we run the existing method on words as per our method to identify the scripts at the word level. The classification rates for both the proposed method and existing method on English and Chinese are reported in Table III where the proposed method gives a better classification rate compared to the existing method. The reason for poorer results of the existing method is that the extracted features are not good enough to handle broken segments and these features expect some regularity of text patterns in each zone. The proposed method is capable of overcoming these problems because the dominant pixel selection and their angular features preserve the uniqueness of scripts in spite of broken segments. On top of this, our method works well for six other scripts. Thus the proposed method is superior to the existing method in terms of classification rate and number of scripts.

TABLE III. PERFORMANCE OF THE PROPOSED METHOD AND EXISTING METHODS AT WORD LEVEL (IN %)

Scripts	Proposed method		Gllavata and Freisleben [1]		
	Confusion matrix		Confusion matrix		
	English	Chinese	Scripts	English	Chinese
English	<b>96.4</b>	5.8	English	68.3	44.0
Chinese	3.6	<b>94.1</b>	Chinese	31.7	56.0

#### IV. CONCLUSION AND FUTURE WORK

We have proposed new Gradient-Angular-Features (GAF) for identifying six scripts. The dominant text pixel selection is performed based on the histograms of the horizontal gradient and vertical gradient division. The eight features which include both local and global information of word images are proposed to identify the scripts. The experimental results show that the GAF features are good enough to tackle the causes of video problems. The performance of the proposed method is compared with an

existing method to show that the proposed method is superior to the existing method in terms of classification rate and applicability. We are planning to extend this method to more scripts using spatio-temporal information for real-time applications.

#### REFERENCES

- [1] J. Gllavata and B. Freisleben, "Script Recognition in Images with Complex Backgrounds", In Proc. IEEE International Symposium on Signal Processing and Information Technology, 2005, pp 589-594.
- [2] D. Doermann, J. Liang and H. Li, "Progress in Camera-Based Document Image Analysis", In Proc. ICDAR, 2003, pp 606-616.
- [3] J. Zang and R. Kasturi, "Extraction of Text Objects in Video Documents: Recent Progress", In Proc. DAS, 2008, pp 5-17
- [4] P. Shivakumara, T. Q. Phan and C. L. Tan, "A Laplacian Approach to Multi-Oriented Text Detection in Video", IEEE Trans. PAMI, 2011, pp 412-419.
- [5] D. Chen and J. M. Odobez, "Video text recognition using sequential Monte Carlo and error voting methods", Pattern Recognition Letters, 2005, pp 1386-1403.
- [6] T. Q. Phan, P. Shivakumara, Z. Ding, S. Lu and C. L. Tan, "Video Script Identification based on Text Lines", In Proc. ICDAR, 2011, pp 1240-1244.
- [7] D. Zhao, P. Shivakumara, S. Lu and C. L. Tan, "New Spatial-Gradient-Features for Video Script Identification", In Proc. DAS, 2012, pp 38-42.
- [8] D. Ghosh, T. Dube and A. P. Shivaprasad, "Script Recognition-Rview", IEEE Trans. PAMI, 2010, pp 2142-2161.
- [9] T. N. Tan, "Rotation Invariant Texture Features and Their Use in Automatic Script Identification", IEEE Transactions on PAMI, 1998, pp 751-756.
- [10] A. Busch, W. W. Boles and S. Sridharan, "Texture for Script Identification", IEEE Trans. PAMI, 2005, pp 1720-1732.
- [11] M. A. Ferrer, A. Morales and U. Pal, "LBP based Line-Wise Script Identification", In Proc. ICDRA, 2013, pp 369-373.
- [12] L. Shijian and C. L. Tan, "Script and Language Identification in Noisy and Degraded Document Images", IEEE Trans. PAMI, 2008, pp 14-24.
- [13] S. Lu, L. Li and C. L. Tan, "Identification of scripts and orientations of degraded document images", Pattern Analysis and Applications, 2010, pp 469-475.
- [14] P. B. Pati and A. G. Ramakrishnan, "Word level multi-script identification", Pattern Recognition Letters, 2008, pp 1218-1229.
- [15] S. Chanda, S. Pal, K. Franke and U. Pal, "Two-stage Approach for Word-wise Script Identification", In Proc. ICDAR, 2009, pp 926-930.
- [16] S. Jaeger, H. Ma and D. Doremann, "Identifying Script on Word-Level with Informational Confidence", In Proc. ICDAR, 2005, pp 416-420.
- [17] L. Li and C. L. Tan, "Script Identification of Camera-based Images", In Proc. ICPR, 2008.
- [18] S. Chanda, O. R. Terrades and U. Pal, "SVM based Scheme for Thai and English Script Identification", In Proc. ICDAR, 2007, pp 551-555.
- [19] S. Ghosh and B. B. Chaudhuri, "Composite Script Identification and Orientation Detection for Indian Text Images", In Proc. ICDAR, 2011, pp 294-298.
- [20] N. Sharma, S. Chanda, U. Pal and M. Blumenstein, "Word-Wise Script Identification from Video Frames", In Proc. ICDAR, 2013, pp 867-871.
- [21] P. P. Roy, U. Pal and J. Liados, "Document seal detection using GHT and character proximity graphs", Pattern Recognition, 2011, pp 1282-1295.
- [22] N. Sharma, P. Shivakumara, U. Pal, M. Blumenstein and C. L. Tan, "A New Method for Words Segmentation from Arbitrarily-Oriented Video Text Lines", In Proc. DICTA, 2012.